

LIS Verification Toolkit (LVT) User's Guide

May 13, 2010

Revision 1.0

History:

Revision	Summary of Changes	Date
1.0	Initial version for LIS 6.0	August, 2009



National Aeronautics and Space Administration
Goddard Space Flight Center
Greenbelt, Maryland 20771

Contents

1	Introduction	4
1.1	What's New	4
1.1.1	Version 1.0	4
2	Background	5
2.1	LVT	5
3	Preliminaries	6
4	Obtaining the Source Code	7
4.1	Downloading the Source Code	7
5	Building the Executable	8
5.0.1	Development Tools	8
5.0.2	Required Software Libraries	8
5.0.3	Optional Software Libraries	8
5.0.4	Build Instructions	9
5.1	Generating documentation	11
6	Running the Executable	12
7	LVT config File	13
7.1	Overall driver options	13
7.2	Runtime options	16
7.3	Domain specification	19
7.3.1	Cylindrical lat/lon	19
7.3.2	Cylindrical lat/lon	19
7.4	Analysis options specification	22
7.5	Observation sources	27
7.5.1	LIS LSM output as the observation	28
7.5.2	CEOP station observations	28
7.5.3	SURFRAD radiation observations	29
7.5.4	SCAN soil moisture observations	29
7.5.5	COOP snow depth observations	30
7.5.6	Walnut Gulch PBMR soil moisture observations	32
7.5.7	SNOTEL SWE observations	32
7.5.8	GSOD snow depth observations	33
7.5.9	LSWG Tb observations	33
7.5.10	FMI SWE observations	35
7.5.11	CMC's daily snow depth observations	35
7.5.12	SNODAS snow analysis data	35
7.6	DA diagnostics analysis	36
7.7	GA output analysis	36
A	Cylindrical Lat/Lon Domain Example	38

B Polar Stereographic Domain Example	43
C Gaussian Domain Example	44
D Lambert Conformal Domain Example	48
E Mercator Domain Example	49
F UTM Domain Example	50
G configure.lvt	51

DRAFT

1 Introduction

This is the User's Guide for LIS verification toolkit (LVT). This document describes how to download and install LVT codes and instructions on building an executable.

This document consists of several sections, described as follows:

- 1 Introduction:** the section you are currently reading
- 2 Background:** general information about the LVT
- 3 Preliminaries:** general information, steps, instructions, and definitions used throughout the rest of this document
- 4 Obtaining the Source Code:** the steps needed to download the source code
- 5 Building the Executable:** the steps needed to build the LVT executable

1.1 What's New

1.1.1 Version 1.0

1. This is the initial version developed for evaluating output from LIS version 6.0 or higher.

2 Background

This section provides some general information about the LVT

2.1 LVT

Insert text here

DRAFT

3 Preliminaries

This section provides some preliminary information to make reading this guide easier.

Commands are written like this:

```
% cd /path/to/LVT
```

```
% ls
```

“...compiler flags, the run `gmake`.”

File names are written like this:

```
/path/to/LVT/src
```

You need to create a working directory on your system to install LVT. Let's call this directory */path/to/LVT/*. Throughout the rest of this document, this directory shall be referred to as *\$WORKING*. You should create a directory to run LVT in. This directory can be created anywhere on your system, but, in this document, we shall refer to this running directory as *\$WORKING/run*.

DRAFT

4 Obtaining the Source Code

This section describes how to obtain the source code needed to build the LVT executable.

The source code is maintained in a Subversion repository; thus, you need the Subversion client (`svn`) to obtain this code. If you need any help regarding Subversion, please go to <http://subversion.tigris.org/>.

4.1 Downloading the Source Code

To obtain the source code needed for LVT, you must first obtain access to the LIS source code repository. To obtain access you must contact the LIS team. Once you have access to the repository, you will be given the correct Subversion command to run to checkout the source code.

Please note that LIS' web-site is under reconstruction.

1. Go to the working directory
`% cd $WORKING`
2. Check out the LIS source code into a directory called `src` under the `$WORKING` directory.
`% svn checkout https://flood.gsfc.nasa.gov/svn/tools/lvt/ src`

5 Building the Executable

This section describes how to build the source code and create LVT executable

5.0.1 Development Tools

This code has been compiled and run on Linux PC (Intel/AMD based) systems, IBM AIX systems, and SGI Altix systems. These instructions expect that you are using such a system. In particular you need

- Linux
 - Absoft’s Pro Fortran Software Development Kit, version 10.0
or
Lahey/Fujitsu’s Fortran 95 Compiler, release L6.00c
 - GNU’s C and C++ compilers, gcc and g++, version 3.3.3
 - GNU’s make, gmake, version 3.77
- IBM
 - XL Fortran version 10.1.0.6
 - GNU’s make, gmake, version 3.77
- SGI Altix
 - Intel Fortran Compiler version 11.1.038
 - GNU’s make, gmake, version 3.77

5.0.2 Required Software Libraries

In order to build the LVT executable, the following libraries must be installed on your system:

- Earth System Modeling Framework (ESMF) version 3.1.0rp2 (<http://www.esmf.ucar.edu/>)
Please read the ESMF User’s Guide for details on compiling ESMF with MPI support and without MPI support (“mpiuni”). Note that ESMF must be compiled with MPI support for using LIS-WRF system in a multiprocessor environment.

5.0.3 Optional Software Libraries

The following libraries are not required to compile LVT. They are used to extend the functionality of LVT.

- Message Passing Interface (MPI) – If you wish to run the MPI-based running mode
 - vendor supplied, or

– MPICH version 1.2.7p1 (<http://www-unix.mcs.anl.gov/mpi/mpich1/>)

- If you choose to have NETCDF support, please download the netcdf source (<http://www.unidata.ucar.edu/software/netcdf/>) and compile the source to generate the NETCDF library.

To install these libraries, follow the instructions provided at the various URL listed above.

5.0.4 Build Instructions

1. Perform the steps described in Section 4 to obtain the source code.
2. Goto the $\$WORKING/src/arch$ directory. A number of files named *configure.lvt.** exist in this directory. Each file contains the configurable options that are specific for each architecture and compiler. For example, the file *configure.lvt.aix* contains the set of configurable options for an IBM AIX platform. Depending on your choice of platform, edit this file or create a new file for your platform with the set of options. The following is a list of variables that need to be specified in the *configure.lvt* file.

Variable	Description
FC	fortran 90 compiler
FC77	fortan 77 compiler
LD	fortran linker
CC	C compiler
AR	program to create a library archive
INC_NETCDF	path to NETCDF header files
LIB_NETCDF	path to NETCDF library files
INC_HDF	path to HDF header files
LIB_HDF	path to HDF library files
INC_HDFEOS	path to HDFEOS header files
LIB_HDFEOS	path to HDFEOS library files
INC_CRTM	path to CRTM header files
LIB_CRTM	path to CRTM library files
LIB_MPI	path to mpi libraries
INC_MPI	path to mpi header files
LIB_ESMF	path to esmf libraries
MOD_ESMF	path to esmf modules
CFLAGS	flags for C compiler
FFLAGS	flags for Fortran 90 compiler
FFLAGS77	flags for Fortran 77 compiler
LDFLAGS	flags for linker

If the user choses to compile and run on a single processor with no MPI, the options in the *configure.lvt* file should be specified accordingly. Specifying the compiler preprocessor flag *-DSPMD* enables the compiling of the code with MPI support. Removing this flag produces a serial version of LVT.

3. Compile the new GRIB library, *libw3.a*. You must edit the *Makefile* located in *\$WORKING/src/lib/w3lib*. Uncomment the appropriate block of compiler flags, then run **gmake**.
4. Compile the new GRIB library, *griblib.a*. You must edit the *Makefile* located in *\$WORKING/src/lib/grib*. Uncomment the appropriate block of compiler flags, then run **gmake**.
5. Compile the new GRIB library, *read_grib*. You must edit the *Makefile* located in *\$WORKING/src/lib/read_grib*. Uncomment the appropriate block of compiler flags, then run **gmake <arch>**, where **<arch>** is the appropriate architecture or compiler label. Running **gmake** will produce a list of acceptable values. For example, to compile on a system using the Intel Fortran compiler, run **gmake ifc**. Please refer to the Appendix ?? for helpful suggestions and instructions. If you are on an IBM AIX system, use the *read_grib.aix* library.
6. All the included libraries are generated. Copy the appropriate *configure.lvt.** file to *\$WORKING/src/make/configure.lvt* and edit this *configure.lvt* file to make sure the file paths are specified correctly.
7. Compile the dependency generator, *makdep*. Change directory into *\$WORKING/src/make/MAKDEP*. Run **gmake**.
8. Compile the LVT source code.
 - (a) Change directory into *\$WORKING/src/make*.

```
% cd $WORKING/src/make
```
 - (b) Edit the *misc.h* file to specify if NETCDF support should be included.
 If **define** *USE_NETCDF* is set, NETCDF support will be included.
 To disable NETCDF support, edit the *misc.h* file to specify **UNDEF** *USE_NETCDF*.
 - (c) Edit the *misc.h* file to specify if HDF4 support should be included.
 If **define** *USE_HDF4* is set, HDF4 support will be included.
 To disable HDF4 support, edit the *misc.h* file to specify **UNDEF** *USE_HDF4*.
 - (d) Edit the *misc.h* file to specify if HDF5 support should be included.
 If **define** *USE_HDF5* is set, HDF5 support will be included.
 To disable HDF5 support, edit the *misc.h* file to specify **UNDEF** *USE_HDF5*.
 - (e) Run the make command.

```
% gmake
```
 - (f) Finally, copy the *LVT* executable into your running directory, *\$WORKING/run*.

See Appendix G to see a *configure.lvt* file.

5.1 Generating documentation

LVT code uses the ProTex documenting system [1]. The documentation in \LaTeX format can be produced by using the `doc.csh` in the `$WORKING/src/utls` directory. This command produces documentation, generating a number of \LaTeX files. These files can be easily converted to pdf or html formats using utilites such as `pdflatex` or `latex2html`.

DRAFT

6 Running the Executable

This section describes how to run the LVT executable.

The single-process version of LVT is executed by the following command issued in the *\$WORKING/run/* directory.

```
% ./LVT <configfile>
```

where *< configfile >* represents the file containing the run time configuration options for LVT.

The parallel version of LVT must be run through an *mpirun* script or similar mechanism. Assuming that MPI is installed correctly, the LVT simulation is carried out by the following command issued from in the *\$WORKING/run/* directory.

```
% mpirun -np N ./LVT <configfile>
```

The *-np N* flag indicates the number of processes to use in the run, where you replace *N* with the number of processes to use. On a multiprocessor machine, the parallel processing capabilities of LVT can be exploited using this flag.

To customize your run, you must specify a LVT runtime configuration file. See Section 7 for more information.

DRAFT

7 LVT config File

This section describes the options in the *lvt.config* file.

7.1 Overall driver options

LIS Running mode: specifies the running mode to be used. Acceptable values are:

Value	Description
1	standard analysis mode
2	data assimilation output analysis
3	data assimilation observation analysis
6	parameter estimation/uncertainty output analysis
7	RTM (radiative transfer model) output analysis

LIS Running mode: 1

LIS Domain type: specifies the “LIS domain” used in the LIS simulation. Acceptable values are:

Value	Description
1	Lat/Lon projection with SW to NE data ordering
2	Mercator projection with SW to NE data ordering
3	Lambert conformal projection with SW to NE data ordering
4	Gaussian domain
5	Polar stereographic projection with SW to NE data ordering
6	AFWA lat/lon 0.5 degree/0.25 degree domain with no subgrid tiling
7	UTM domain
8	HRAP domain
10	Catchment based domain
11	GSWP domain

LIS Domain type: 1

LIS nest index: specifies the nest index of the LIS output being compared using LVT

LIS nest index: 1

Land surface model: specifies the land surface model used in the LIS simulation. Acceptable values are:

Value	Description
0	template lsm
1	Noah 2.7.1
2	CLM 2.0
3	VIC
4	mosaic
5	hyssib
6	sib2
7	catchment
8	sacramento
9	snow17
10	sacramento+snow17
11	sib3
12	mlbc
13	csu
14	place
21	noah 3.1

Land surface model: 1

LIS Output format: specifies the format of the LIS output data. Acceptable values are:

Value	Description
1	LIS output in binary format
2	LIS output in Grib format
3	LIS output in NETCDF format

LIS Output format: 1

LIS Output format: specifies the style of the model output names and their directory organization. Acceptable values are:

Value	Description
1	5 levels of hierarchy
2	3 levels of hierarchy
3	2 levels of hierarchy
4	WMO convention for weather codes

LIS Output naming style: 1

Map projection of parameter data: specifies the map projection of the parameter datasets. Note that the grid description options for the parameters will be different for different map projections

Acceptable values are:

Value	Description
0	Equidistant cylindrical (lat/lon)
4	Gaussian
5	Polar Stereographic
6	UTM projection

Map projection of parameter data: 0

Number of observation sources: specifies the number of observational data sources to be used in comparisons

Number of observation sources: 1

Observation source: specifies the observational data to be used for comparing LIS model output

Acceptable values are:

Starting day:	1
Starting hour:	0
Starting minute:	0
Starting second:	0

The end time of the evaluation period is specified in the following format:

Variable	Value	Description
Ending year:	integer 2001 – present	specifying ending year
Ending month:	integer 1 – 12	specifying ending month
Ending day:	integer 1 – 31	specifying ending day
Ending hour:	integer 0 – 23	specifying ending hour
Ending minute:	integer 0 – 59	specifying ending minute
Ending second:	integer 0 – 59	specifying ending second

Ending year:	2008
Ending month:	5
Ending day:	31
Ending hour:	0
Ending minute:	0
Ending second:	0

LVT Output format: specifies the format of the LVT output. Acceptable values are:

Value	Description
1	Write output in binary format
2	Write output in Grib format (not supported yet)
3	Write output in NETCDF format

LVT Output format:	3
--------------------	---

LIS Output timestep: specifies the frequency of model outputs used in the LIS simulation (in seconds)

LIS Output timestep:	10800
----------------------	-------

Undefined value: specifies the undefined value. The default is set to -9999.

Undefined value: -9999

LVT diagnostic file: specifies the name of run time diagnostic file. Acceptable values are any 40 character string.

LVT diagnostic file: lvtlog

LIS Output methodology: specifies the output methodology used in the LIS simulation. The LIS output is written as a 1-D array containing only land points or as a 2-D array containing both land and water points. 1-d tile space includes the subgrid tiles and ensembles. 1-d grid space includes a vectorized, land-only grid-averaged set of values. Acceptable values are:

Value	Description
1	LIS output in a 1-D tile domain
2	LIS output in a 2-D grid domain
3	LIS output in a 1-D grid domain

LIS Output methodology: 2

LIS Output directory: specifies the name of the top-level LIS output directory. Acceptable values are any 40 character string. For simplicity, throughout the rest of this document, this top-level output directory shall be referred to by its default name, *\$WORKING/LIS/OUTPUT*.

LIS Output directory: ./CTRL/OUTPUT

Number of ensembles per tile: specifies the number of ensembles of tiles used in the LIS simulation. The value should be greater than or equal to 1.

Number of ensembles per tile: 1

This section specifies the 2-d layout of the processors in a parallel processing environment. The user can specify the number of processors along the east-west dimension and north-south dimension using **Number of processors along x:**

and `Number of processors along y:`, respectively. Note that the layout of processors should match the total number of processors used. For example, if 8 processors are used, the layout can be specified as 1x8, 2x4, 4x2, or 8x1.

```
Number of processors along x:  2
Number of processors along y:  2
```

7.3 Domain specification

This section of the config file specifies the running domain (domain over which the simulation is carried out). The specification of the running domain section depends on the type of LIS domain and projection used. Section 7.1 lists the projections that LIS supports.

7.3.1 Cylindrical lat/lon

This section describes how to specify a cylindrical latitude/longitude projection. See Appendix A for more details about setting these values.

```
run domain lower left lat:      30.125
run domain lower left lon:     -124.875
run domain upper right lat:    50.125
run domain upper right lon:   -69.875
run domain resolution (dx):    0.25
run domain resolution (dy):    0.25
```

#Definition of Parameter Domain

7.3.2 Cylindrical lat/lon

This section describes how to specify a cylindrical latitude/longitude projection. See Appendix A for more details about setting these values.

```
param domain lower left lat:   -59.875
param domain lower left lon:  -179.875
param domain upper right lat:   89.875
param domain upper right lon:  179.875
param domain resolution (dx):   0.25
```

param domain resolution (dy): 0.25

Number of veg types: specifies the number of vegetation types used in the landcover data, used in the LIS simulation. Acceptable values are:

Value	Description
13	UMD-based landcover types
16	IGBP-based landcover types
30	USGS-based landcover types

Number of veg types: 24

The following options are used for subgrid tiling based on vegetation

Maximum number of tiles per grid: defines the maximum tiles per grid (this can be as many as the total number of vegetation types), used in the LIS simulation

Maximum number of tiles per grid: 1

Cutoff percentage: defines the smallest percentage of a cell for which to create a tile, used in the LIS simulation. The percentage value is expressed as a fraction.

Cutoff percentage: 0.05

Landcover data source: specifies the usage of landcover data in the LIS run. Acceptable values are:

Value	Description
1	UMD landcover
2	USGS landcover data
3	GFS landcover data
4	IGBP landcover data

landcover data source: 2

landmask file: specifies the location of land/water mask file.
landcover file: specifies the location of the vegetation classification file
landcover file format: specifies if the vegetation file is tiled or not (0-not tiled, 1- tiled)

```
landmask file:                ./input/AFWA-25KM/mask_25KM.1gd4r  
landcover file:              ./input/UMD-25KM/usgs_veg_25km.1gd4r  
landcover file format:      0
```

This section should also specify the domain specifications of the landcover data. If the map projection of parameter data is specified to be lat/lon, the following configuration should be used for specifying landcover data See Appendix A for more details about setting these values.

```
landcover lower left lat:    -59.875  
landcover lower left lon:   -179.875  
landcover upper right lat:   89.875  
landcover upper right lon:  179.875  
landcover resolution (dx):   0.25  
landcover resolution (dy):   0.25
```

LIS output attributes file: specifies the output attributes file. This file can be specified by using the model output attributes used to for customizing the LIS model output. An extra column needs to be added in this file to specify which variables among the LIS output are to be included in the evaluation/verification.

```
LIS output attributes file:  './MODEL_OUTPUT_LIST_LVT.TBL'
```

LIS soil moisture layer thickness: specifies the thickness values of the soil moisture layers

```
LIS soil moisture layer thickness:    100 300 600 1000
```

LIS soil temperature layer thickness: specifies the thickness values of the soil temperature layers

LIS soil temperature layer thickness: 100 300 600 1000

7.4 Analysis options specification

This section of the config file specifies the type of analysis to be conducted during the verification/evaluation. Note that some options are only available in certain running modes.

Apply external mask: Specifies whether to apply an external mask in limiting the analysis to a selected set of data points

Acceptable values are:

Value	Description
0	Do not apply external mask
1	Apply external, temporally varying mask
2	Apply fixed mask

Apply external mask: 0

External mask directory: Specifies the name of the data mask file/directory. If the mask varies temporally, then this option specifies the top-level directory containing data mask. Note that the mask files are expected to be in binary, sequential access format.

External mask directory: none

Compute Mean (Total): Specifies if temporal mean values of LIS output (for the entire evaluation period) are to be computed.

Acceptable values are:

Value	Description
0	Do not compute
1	Compute

Compute Mean (Total): 0

Compute Mean (Time Series): Specifies if temporal mean values of LIS output (based on the frequency of statistics output) are to be computed.

Acceptable values are:

Value	Description
0	Do not compute
1	Compute

Compute Mean (Time Series): 0

Compute Std: Specifies if standard deviation values of LIS output (based on the specified time period) are to be computed.

Acceptable values are:

Value	Description
0	Do not compute
1	Compute

Compute Std: 0

Compute RMSE (Total): Specifies if RMSE values for the entire evaluation time period are to be computed.

Acceptable values are:

Value	Description
0	Do not compute
1	Compute

Compute RMSE (Total): 1

Compute Bias (Total): Specifies if Bias values for the entire evaluation time period are to be computed.

Acceptable values are:

Value	Description
0	Do not compute
1	Compute

Compute Bias (Total): 1

Compute RMSE (Time Series): Specifies if a time series of RMSE values during the evaluation period are to be computed.

Acceptable values are:

Value	Description
0	Do not compute
1	Compute

Compute RMSE (Time Series): 1

Compute Bias (Time Series): Specifies if a time series of Bias values during the evaluation period are to be computed.

Acceptable values are:

Value	Description
0	Do not compute
1	Compute

Compute Bias (Time Series): 1

Compute Anomaly Correlations: Specifies if anomaly time series correlation coefficient values (for the entire evaluation period) are to be computed. Note that if masking is turned on, the code will use only "unmasked" data points for computations.

Acceptable values are:

Value	Description
0	Do not compute
1	Compute

Compute Anomaly Correlations: 0

Compute Raw Correlations: Specifies if raw time series correlation coefficient values (for the entire evaluation period) are to be computed. Note that if masking is turned on, the code will use only "unmasked" data points for computations.

Acceptable values are:

Value	Description
0	Do not compute
1	Compute

Compute Raw Correlations: 0

Compute POD (Total): Specifies if probability of detection values for the entire evaluation time period are to be computed.

Acceptable values are:

Value	Description
0	Do not compute
1	Compute

Compute POD (Total): 1

Compute POD (Time Series): Specifies if a time series of Probability of detection (POD) values (based on the frequency of statistics output) during the evaluation period are to be computed.

Acceptable values are:

Value	Description
0	Do not compute
1	Compute

Compute POD (Time Series): 1

Compute FAR (Total): Specifies if false alarm ratio values for the entire evaluation time period are to be computed.

Acceptable values are:

Value	Description
0	Do not compute
1	Compute

Compute FAR (Total): 1

Compute FAR (Time Series): Specifies if a time series of false alarm ratio (FAR) values (based on the frequency of statistics output) during the evaluation period are to be computed.

Acceptable values are:

Value	Description
0	Do not compute
1	Compute

Compute FAR (Time Series): 1

Compute Peak Detection: Specifies if the time of occurrence of peak values of variables are to be computed. The code will output the peak occurrence times of the model values and the observations

Acceptable values are:

Value	Description
0	Do not compute
1	Compute

Compute Peak Detection: 1

Observation count threshold: Specifies the number of observations to be used as the minimum threshold for computing statistics. Grid points with observation count less than this value will be ignored.

Acceptable values are 0 or higher

Observation count threshold: 0

Temporal averaging interval: Specifies (in seconds) the temporal averaging interval of the LIS output and observation data.

Temporal averaging interval: 86400

Stats output directory: Specifies the top-level directory where the output from the analysis is to be written.

Stats output directory: ./STATS

Stats output interval: Specifies the frequency (in seconds) of the analysis output

Stats output interval: 86400

Extract time series: Specifies if an ASCII time series file of model output and observations are to be written, for specified locations in the domain.

Extract time series: 1

Time series location file: Specifies the name of the file which lists the locations in the domain where the time series data are to be derived. The format of the time series location file is as follows:

```
#Number of locations
2
#Location style (1-lat/lon, 2-col/row, 3-tile)
1
#Location name (location 1)
WEST_US
#lat/lon values
40 -130 50 -110
#Location name (location 2)
HIGH_PLAINS_US
#lat/lon values
43 -110 49 -100
```

Time series location file: ./ts_locations.txt

7.5 Observation sources

This section of the config file specifies the details of the observational sources.

7.5.1 LIS LSM output as the observation

LISlsmObs model name: name of the model used in the simulation. Acceptable values are:

Value	Description
NOAH	Noah land surface model
CLSM	Catchment land surface model

LISlsmObs number of variables in output: specifies the number of simulated variables in the output

LISlsmObs domain type: specifies the domain type used in generating the output

LISlsmObs nest index: specifies the nest index of the domain

LISlsmObs experiment code: specifies the experiment code number used in the simulation

The domain should be specified, based on the domain type. LISlsmObs model output attributes file: specifies the model output attribute file used for generating the output

```
LISlsmObs model name:      CLSM
LISlsmObs number of variables in output:  23
LISlsmObs domain type:      1
LISlsmObs nest index:      1
LISlsmObs experiment code:  111
LISlsmObs output directory:  ./CLSM/OUTPUT
LISlsmObs domain lower left lat:  30.5
LISlsmObs domain lower left lon: -124.5
LISlsmObs domain upper right lat:  50.5
LISlsmObs domain upper right lon: -75.5
LISlsmObs domain resolution (dx):  1.0
LISlsmObs domain resolution (dy):  1.0
LISlsmObs model output attributes file:  './CLSM_OUTPUT_LIST.TBL'
```

7.5.2 CEOP station observations

CEOP undefined value: specifies the undefined value used in CEOP data

CEOP metadata file: specifies the file that lists the metadata for the CEOP stations.

CEOP read surface meteorology data: specifies whether to read the surface meteorology data (.true. or .false.)

CEOP read flux data: specifies whether to read the surface fluxes data (.true. or .false.)

CEOP read soil moisture and temperature data: specifies whether to read the soil moisture and temperature data (.true. or .false.)

CEOP soil moisture layer weights: specifies the vertical interpolation weights for soil moisture

CEOP soil temperature layer weights: specifies the vertical interpolation weights for soil temperature

CEOP surface meteorology data file: specifies the surface meteorology data file

CEOP flux data file: specifies the flux data file

CEOP soil temperature and moisture data file: specifies the soil temperature and moisture data file.

CEOP undefined value: -999.99

CEOP metadata file: ./bon.mdata

CEOP read surface meteorology data: .true.

CEOP read flux data: .true.

CEOP read soil moisture and temperature data: .true.

CEOP soil moisture layer weights: 3 2 6 10 22 42

CEOP soil temperature layer weights: 3 2 6 10 22 42

CEOP surface meteorology data file: ./CEOP/GAPP/GAPP_Bondville_20021001_20041231.sfc

CEOP flux data file: ./CEOP/GAPP/GAPP_Bondville_20021001_20041231.flx

CEOP soil temperature and moisture data file: ./CEOP/GAPP/GAPP_Bondville_20021001_20041231.s

7.5.3 SURFRAD radiation observations

SURFRAD observation directory: specifies the location of the SURFRAD radiation data

SURFRAD observation directory: ../SURFRAD

7.5.4 SCAN soil moisture observations

SCAN observation directory: specifies the location of the SCAN soil moisture observation data

SCAN coord file: specifies the file that lists the location of the SCAN stations
SCAN metadata file: specifies the file that lists the metadata for the SCAN stations.

SCAN soil moisture layer weights: specifies the vertical interpolation weights for soil moisture
SCAN soil temperature layer weights: specifies the vertical interpolation weights for soil temperature

```
SCAN observation directory:      ./SCAN/
SCAN coord file:                 ./SCAN_coord.txt
SCAN metadata file:              ./SCAN_mdata
SCAN soil moisture layer weights: 0.075 0.075 0.205 0.460 0.185
SCAN soil temperature layer weights: 0.075 0.075 0.205 0.460 0.185
```

7.5.5 COOP snow depth observations

COOP observation directory: specifies the location of the COOP snow depth observation data

COOP coord file: specifies the file that lists the location of the COOP stations.
The format of the station list is as follows:

```
010008      ABBEVILLE                31.570      -84.752
010116      ALABASTER SHELBY CO AP    33.178      -85.218
010140      ALBERTA                   32.232      -86.589
.....
.....
```

COOP metadata file: specifies the file that lists the metadata for the COOP stations. The format of the metadata file is as follows:

```
#nstns udef   syr  smo sda smn  eyr eda emo ehr emn ts
10395 -9999.0 2007 11 01 01 00 2008 06 01 00 00 3600
#nstates
47
#state names
AL
```

AR
AZ
CA
CO
CT
FL
GA
IA
ID
IL
IN
KS
KY
LA
MA
MD
ME
MI
MN
MO
MS
MT
NC
ND
NE
NH
NJ
NM
NV
NY
OH
OK
OR
PA
RI
SC
SD
TN
TX
UT
VA
VT
WA
WI
WV
WY

DRAFT

```

COOP observation directory:      ./COOP
COOP coord file:                ./COOP/COOP_stnlist.dat
COOP metadata file:            ./COOP/COOP_mdata

```

7.5.6 Walnut Gulch PBMR soil moisture observations

WG PBMR observation directory: specifies the location of the Walnut Gulch PBMR soil moisture observation data
WG PBMR site index: specifies the station being used

```

WG PBMR observation directory:  ./WG PBMR
WG PBMR coord file:            1

```

7.5.7 SNOTEL SWE observations

SNOTEL observation directory: specifies the location of the SNOTEL SWE observation data
SNOTEL coord file: specifies the file that lists the location of the SNOTEL stations. The format of the station list is as follows:

AZ	BAKER BUTTE	11R06S	308	34.450	-111.400
AZ	BAKER BUTTE SMT	11R07S	1140	34.450	-111.367
AZ	BALDY	09S01S	310	33.967	-109.500
AZ	BEAVER HEAD	09S06S	902	33.683	-109.200
				
				

SNOTEL metadata file: specifies the file that lists the metadata for the SNOTEL stations. The format of the metadata file is as follows:

```

#nstns, undef, starting time, ending time data, timestep
712 -9999.0 2007 01 01 01 00 2008 12 31 00 00 86400

```

```

SNOTEL observation directory:      ./SNOTEL
SNOTEL coord file:                ./SNOTEL/SNOTEL_CONUS_list.txt
SNOTEL metadata file:             ./SNOTEL/SNOTEL_mdata

```

7.5.8 GSOD snow depth observations

GSOD observation directory: specifies the location of the GSOD snow depth observation data

GSOD coord file: specifies the file that lists the location of the GSOD stations. The format of the station list is as follows:

```

000000 99999 NYGGBUKTA GREENLAND- STA      GL GL          +73483 +021567 +00030
000010 99999 JAN HAYEN                      NO NO          +70983 -007700 +00229
000020 99999 ISFJORD RADIO SPITZBERGEN     NO NO          +78067 +013633 +00079
000030 99999 BJORNOYA BARENTS SEA          NO NO          +74467 +019283 +00290
000040 99999 VAROO                          NO NO          +70367 +031100 +00119
000050 99999 INGOY                          NO NO          +71067 +024150 +00040
.....
.....

```

GSOD metadata file: specifies the file that lists the metadata for the GSOD stations. The format of the metadata file is as follows:

```

#nstns, undef, starting time, ending time data, timestep
30727 -9999.0 2007 11 01 01 00 2008 06 01 00 00 86400

```

```

GSOD observation directory:      ./GSOD
GSOD coord file:                 ./GSOD/GSOD_CONUS_list.txt
GSOD metadata file:              ./GSOD/GSOD_mdata

```

7.5.9 LSWG Tb observations

LSWG Tb observation filename: specifies the name of the LSWG filename containing Brightness Temperature (Tb) observations

LSWG Tb satellite name: specifies the name of satellite – same as what’s used in CRTM

LSWG Tb data format: 0 for AMSR-E, 1-for AMSU LSWG Tb metadata file: specifies the file that lists the metadata for LSWG Tb observations. The format of the metadata file is as follows:

```
#nstns, undef, starting time, ending time, timestep (mins)
1 -1 2006 07 01 10 00 2007 06 30 17 00 3600
#LIS channel data index in file
1 1
2 2
3 3
4 4
5 5
6 6
7 7
8 8
9 9
10 10
11 11
12 12
13 13
14 14
15 15
```

LSWG Tb include cloud masking: specifies if data is to be ignored in the presence of clouds (0-do not ignore, 1-ignore) LSWG Tb cloud mask file: specifies the name of the cloud mask file LSWG Tb cloud mask column: ?? LSWG Tb cloud mask threshold(%): specifies the threshold below which clouds can be ignored (used only if cloud masking is enabled).

```
LSWG Tb observation filename: './_LSWG/C3VP.txt'
LSWG Tb satellite name: 'N18_'
LSWG Tb data format: 1
LSWG Tb metadata file: './C3VP_mdata'
LSWG Tb include cloud masking: 1
LSWG Tb cloud mask file: './cloud_mask.txt'
LSWG Tb cloud mask column: ??
LSWG Tb cloud mask threshold(%): 75
```

7.5.10 FMI SWE observations

FMI observation directory: specifies the location of the FMI snow course data

FMISWE observation directory: ./FMI_SWE

7.5.11 CMC's daily snow depth observations

CMC SNWD observation directory: specifies the location of the CMC snow depth observation data

CMC SNWD metadata file: specifies the file that lists the metadata for the CMC snow depth data. The format of the metadata file is as follows:

```
#undef, starting time, ending time data, timestep  
-9999.0 2007 11 01 00 00 2008 12 30 00 00 86400
```

CMC SNWD observation directory: ./CMC_data

CMC SNWD metadata file: ./CMC_data/CMC SNWD_mdata

7.5.12 SNODAS snow analysis data

SNODAS observation directory: specifies the location of the SNODAS data

SNODAS metadata file: specifies the file that lists the metadata for the SNODAS.

The format of the metadata file is as follows:

```
#undef, starting time, ending time data, timestep  
-9999.0 2007 11 01 00 00 2008 12 30 00 00 86400
```

SNODAS observation directory: ./SNODAS
SNODAS metadata file: ./SNODAS/SNODAS_mdata

7.6 DA diagnostics analysis

This section of the config file specifies the specialized options to analyze the data assimilation diagnostics

Compute Innovation Distribution: Specifies if innovation distribution analysis (computing mean and variance) is to be computed.

Acceptable values are:

Value	Description
0	Do not compute
1	Compute

Compute Innovation Distribution: 0
Compute Analysis Gain: 0
Number of state variables in the DA update: 0

7.7 GA output analysis

This section of the config file specifies the specialized options to analyze Genetic Algorithm output

The following options are for analyzing optimization/uncertainty estimation data output `OptUE algorithm used`: specifies the index of the optimization/uncertainty estimation algorithm used

Acceptable values are:

Value	Description
1	Levenberg- Marquardt
2	Genetic Algorithm
3	SCE-UA
5	MCMC
6	DREAM

OptUE algorithm used: 2

OptUE Decision Space Attributes File: lists the decision space attributes file used in the LIS optimization/uncertainty estimation integration.

OptUE Decision Space Attributes File: ./GARun/noah_sm_decspace.txt

Number of Iterations: Number of generations used in the optimization/uncertainty estimation algorithm.

OptUE Number of Iterations: 20

Compute OptUE time series: specifies if a time series of OptUE run output data is to be generated (0-no, 1-yes)

OptUE Compute time series: 1

OptUE Time series location file: specifies the file which lists the locations in the domain where the time series data are to be extracted. The format of the time series location file is as follows:

```
#Number of locations
1
#Location style (1-lat/lon, 2-col/row, 3-tile)
2
#mask filename
none
#site name
Site1
244 236
```

OptUE Time series location file: ./STN_LOCATIONS.DAT

A Cylindrical Lat/Lon Domain Example

This section describes how to compute the values for the run domain and param domain sections on a cylindrical lat/lon projection.

First, we shall generate the values for the parameter data domain. LIS' parameter data is defined on a Latitude/Longitude grid, from -180 to 180 degrees longitude and from -60 to 90 degrees latitude.

For this example, consider running at $1/4$ deg resolution. The coordinates of the south-west and the north-east points are specified at the grid-cells' centers. Here the south-west grid-cell is given by the box $(-180, -60), (-179.750, -59.750)$. The center of this box is $(-179.875, -59.875)$.¹

```
param domain lower left lat: -59.875
param domain lower left lon: -179.875
```

The north-east grid-cell is given by the box $(179.750, 89.750), (180, 90)$. Its center is $(179.875, 89.875)$.

```
param domain upper right lat: 89.875
param domain upper right lon: 179.875
```

Setting the resolution (0.25 deg) gives

```
param domain resolution dx: 0.25
param domain resolution dy: 0.25
```

And this completely defines the parameter data domain.

Next, we shall generate the values for the running domain.

If you wish to run over the whole domain defined by the parameter data domain then you simply set the values defined in the parameter domain section in the run domain section. This gives:

```
run domain lower left lat: -59.875
run domain lower left lon: -179.875
run domain upper right lat: 89.875
run domain upper right lon: 179.875
run domain resolution dx: 0.25
run domain resolution dy: 0.25
```

Now say you wish to run only over the region given by $(-97.6, 27.9), (-92.9, 31.9)$. Since the running domain is a sub-set of the parameter domain, it is also a Latitude/Longitude domain at $1/4$ deg. resolution. Thus,

```
run domain resolution dx: 0.25
run domain resolution dy: 0.25
```

Now, since the running domain must fit onto the parameter domain, the desired running region must be expanded from $(-97.6, 27.9), (-92.9, 31.9)$ to $(-97.75, 27.75), (-92.75, 32.0)$. The south-west grid-cell for the running domain is the box $(-97.75, 27.75), (-97.5, 28.0)$. Its center is $(-97.625, 27.875)$; giving

¹Note, these coordinates are ordered (longitude, latitude).

```
run domain lower left lat: 27.875
run domain lower left lon: -97.625
```

The north-east grid-cell for the running domain is the box $(-93, 31.75), (-92.75, 32.0)$. Its center is $(-92.875, 31.875)$; giving

```
run domain upper right lat: 31.875
run domain upper right lon: -92.875
```

This completely defines the running domain.

Note, the LIS project has defined 5 km resolution to be 0.05 deg. and 1 km resolution to be 0.01 deg. If you wish to run at 5 km or 1 km resolution, redo the above example to compute the appropriate grid-cell values.

See Figure 1 for an illustration of adjusting the running grid. See Figures 2 and 3 for an illustration of the south-west and north-east grid-cells.

DRAFT

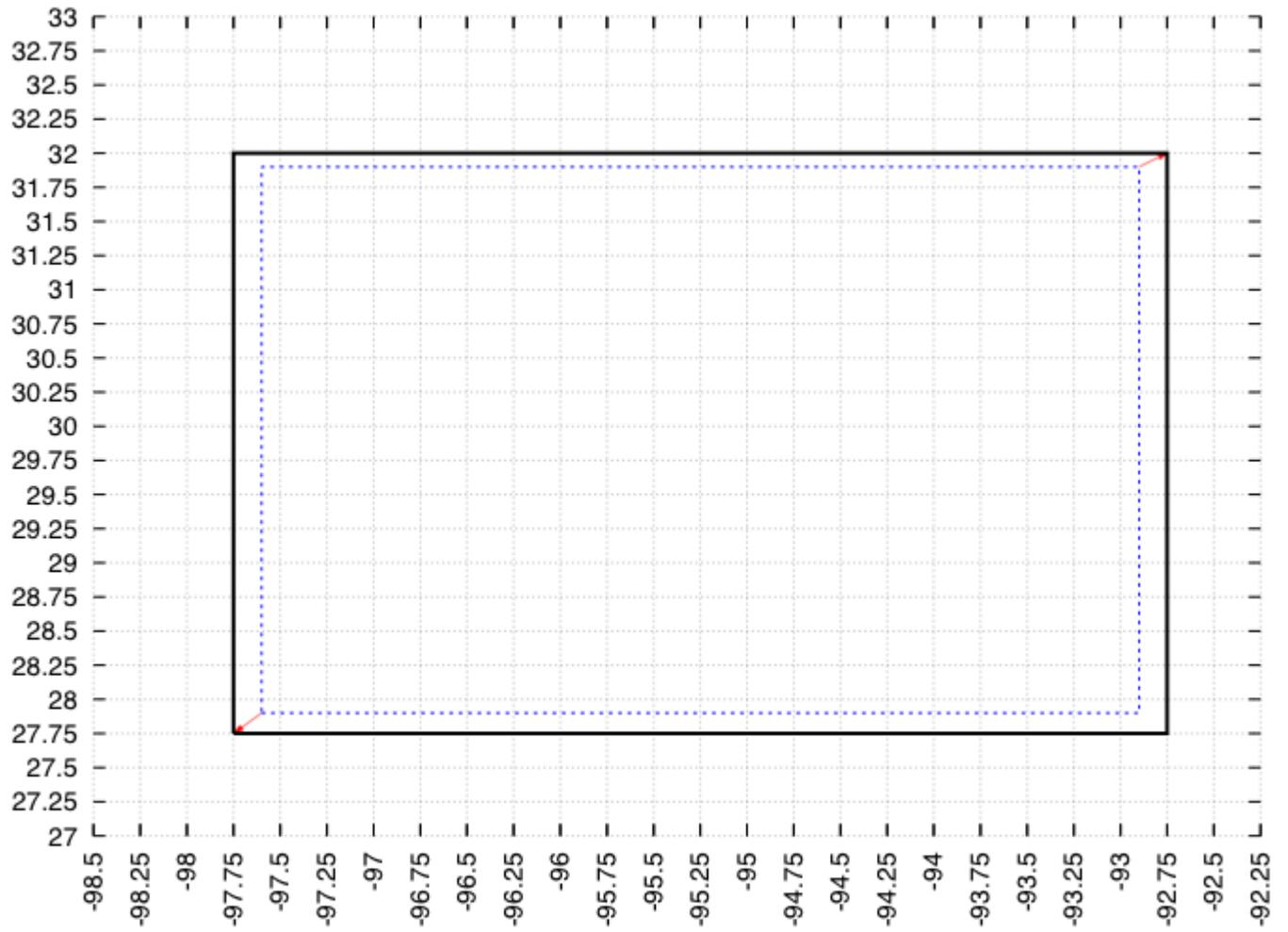


Figure 1: Illustration showing how to fit the desired running grid onto the actual grid

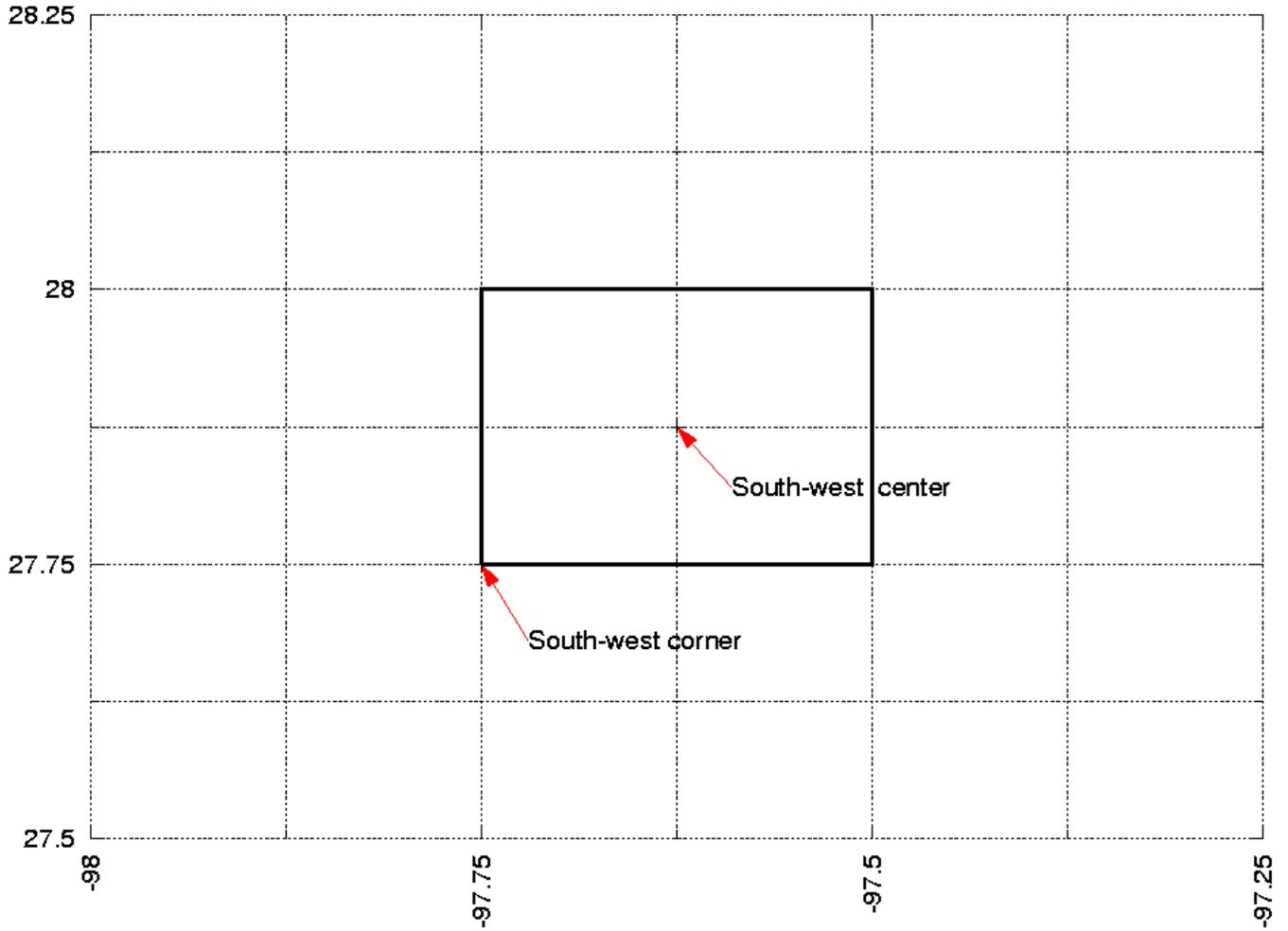


Figure 2: Illustration showing the south-west grid-cell corresponding to the example in Section A

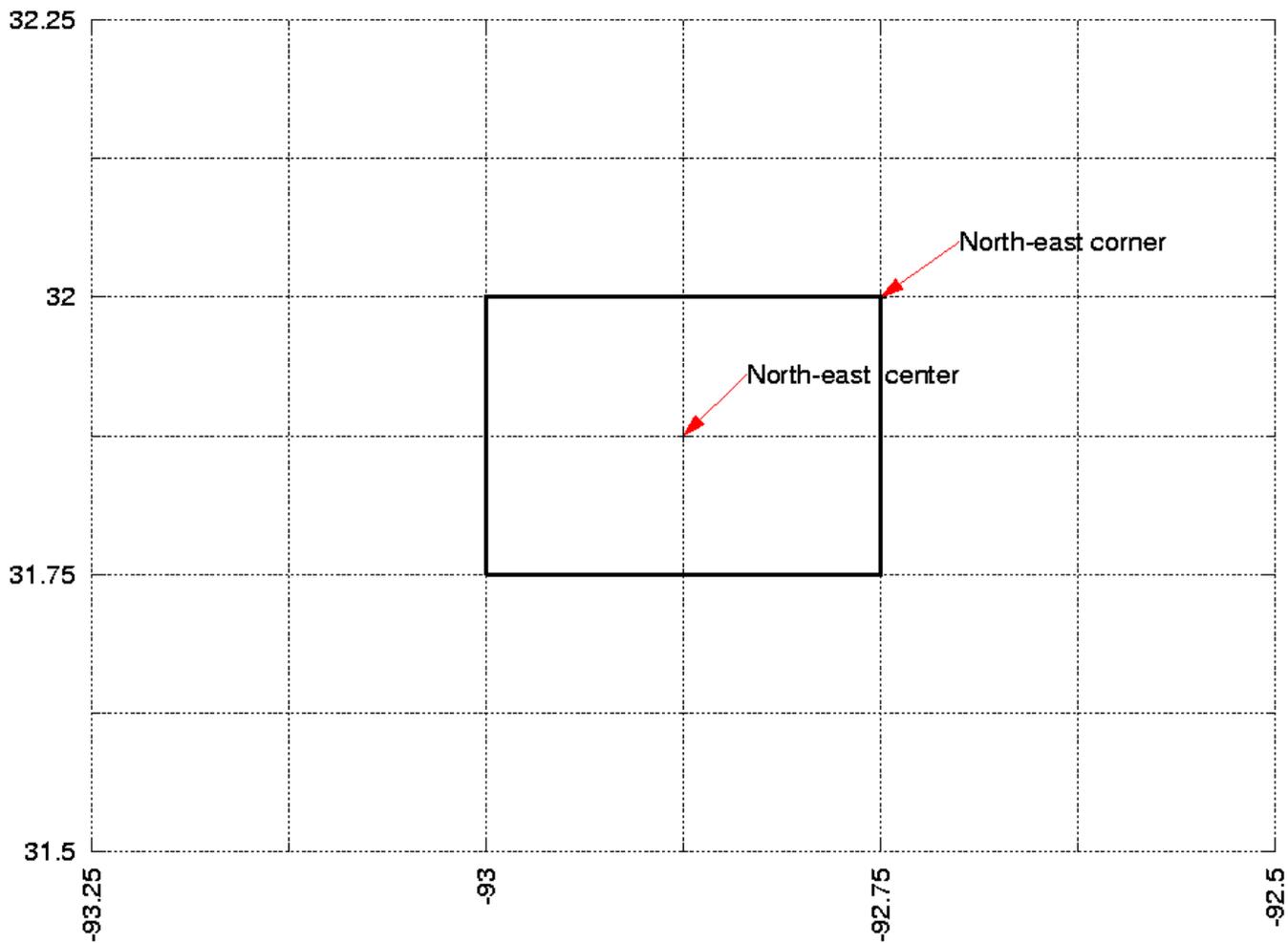


Figure 3: Illustration showing the north-east grid-cell corresponding to the example in Section A

B Polar Stereographic Domain Example

This section describes how to compute the values for the run domain and param domain sections on a polar stereographic projection.

STUB!

DRAFT

C Gaussian Domain Example

This section describes how to compute the values for the run domain and param domain sections on a Gaussian projection.

First, we shall generate the values for the parameter data domain. LIS' Gaussian parameter data is defined from -180 to 180 degrees longitude and from -90 to 90 degrees latitude. Note that the first longitude point is at 0 .

The parameter domain must be specified as follows:

```
param domain first grid point lat:      -89.27665
param domain first grid point lon:       0.0
param domain last grid point lat:       89.27665
param domain last grid point lon:       -0.9375
param domain resolution dlon:           0.9375
param domain number of lat circles:      95
```

Next, we shall generate the values for the running domain.

If you wish to run over the whole domain defined by the parameter data domain then you simply set the values defined in the parameter domain section in the run domain section. This gives:

```
run domain first grid point lat:      -89.27665
run domain first grid point lon:       0.0
run domain last grid point lat:       89.27665
run domain last grid point lon:       -0.9375
run domain resolution dlon:           0.9375
run domain number of lat circles:      95
```

If you wish to run over a sub-domain, then you must choose longitude and latitude values that correspond to the T126 Gaussian projection. Tables of acceptable longitude and latitude values are found below.

Now say you wish to run only over the region given by $(-97.6, 27.9)$, $(-92.9, 31.9)$. Since the running domain must fit on the T126 Gaussian grid, the running domain must be expanded to $(-98.4375, 27.87391)$, $(-91.875, 32.59830)$. Thus the running domain specification is:

```
run domain first grid point lat:      27.87391
run domain first grid point lon:      -98.4375
run domain last grid point lat:       32.59830
run domain last grid point lon:       -91.875
run domain resolution dlon:           0.9375
run domain number of lat circles:      95
```

Table 1: Acceptable longitude values

0.000000	0.937500	1.875000	2.812500	3.750000
4.687500	5.625000	6.562500	7.500000	8.437500
9.375000	10.312500	11.250000	12.187500	13.125000
14.062500	15.000000	15.937500	16.875000	17.812500
18.750000	19.687500	20.625000	21.562500	22.500000
23.437500	24.375000	25.312500	26.250000	27.187500
28.125000	29.062500	30.000000	30.937500	31.875000
32.812500	33.750000	34.687500	35.625000	36.562500
37.500000	38.437500	39.375000	40.312500	41.250000
42.187500	43.125000	44.062500	45.000000	45.937500
46.875000	47.812500	48.750000	49.687500	50.625000
51.562500	52.500000	53.437500	54.375000	55.312500
56.250000	57.187500	58.125000	59.062500	60.000000
60.937500	61.875000	62.812500	63.750000	64.687500
65.625000	66.562500	67.500000	68.437500	69.375000
70.312500	71.250000	72.187500	73.125000	74.062500
75.000000	75.937500	76.875000	77.812500	78.750000
79.687500	80.625000	81.562500	82.500000	83.437500
84.375000	85.312500	86.250000	87.187500	88.125000
89.062500	90.000000	90.937500	91.875000	92.812500
93.750000	94.687500	95.625000	96.562500	97.500000
98.437500	99.375000	100.312500	101.250000	102.187500
103.125000	104.062500	105.000000	105.937500	106.875000
107.812500	108.750000	109.687500	110.625000	111.562500
112.500000	113.437500	114.375000	115.312500	116.250000
117.187500	118.125000	119.062500	120.000000	120.937500
121.875000	122.812500	123.750000	124.687500	125.625000
126.562500	127.500000	128.437500	129.375000	130.312500
131.250000	132.187500	133.125000	134.062500	135.000000
135.937500	136.875000	137.812500	138.750000	139.687500
140.625000	141.562500	142.500000	143.437500	144.375000
145.312500	146.250000	147.187500	148.125000	149.062500
150.000000	150.937500	151.875000	152.812500	153.750000
154.687500	155.625000	156.562500	157.500000	158.437500
159.375000	160.312500	161.250000	162.187500	163.125000
164.062500	165.000000	165.937500	166.875000	167.812500
168.750000	169.687500	170.625000	171.562500	172.500000
173.437500	174.375000	175.312500	176.250000	177.187500
178.125000	179.062500	180.000000	-179.062500	-178.125000

-177.187500	-176.250000	-175.312500	-174.375000	-173.437500
-172.500000	-171.562500	-170.625000	-169.687500	-168.750000
-167.812500	-166.875000	-165.937500	-165.000000	-164.062500
-163.125000	-162.187500	-161.250000	-160.312500	-159.375000
-158.437500	-157.500000	-156.562500	-155.625000	-154.687500
-153.750000	-152.812500	-151.875000	-150.937500	-150.000000
-149.062500	-148.125000	-147.187500	-146.250000	-145.312500
-144.375000	-143.437500	-142.500000	-141.562500	-140.625000
-139.687500	-138.750000	-137.812500	-136.875000	-135.937500
-135.000000	-134.062500	-133.125000	-132.187500	-131.250000
-130.312500	-129.375000	-128.437500	-127.500000	-126.562500
-125.625000	-124.687500	-123.750000	-122.812500	-121.875000
-120.937500	-120.000000	-119.062500	-118.125000	-117.187500
-116.250000	-115.312500	-114.375000	-113.437500	-112.500000
-111.562500	-110.625000	-109.687500	-108.750000	-107.812500
-106.875000	-105.937500	-105.000000	-104.062500	-103.125000
-102.187500	-101.250000	-100.312500	-99.375000	-98.437500
-97.500000	-96.562500	-95.625000	-94.687500	-93.750000
-92.812500	-91.875000	-90.937500	-90.000000	-89.062500
-88.125000	-87.187500	-86.250000	-85.312500	-84.375000
-83.437500	-82.500000	-81.562500	-80.625000	-79.687500
-78.750000	-77.812500	-76.875000	-75.937500	-75.000000
-74.062500	-73.125000	-72.187500	-71.250000	-70.312500
-69.375000	-68.437500	-67.500000	-66.562500	-65.625000
-64.687500	-63.750000	-62.812500	-61.875000	-60.937500
-60.000000	-59.062500	-58.125000	-57.187500	-56.250000
-55.312500	-54.375000	-53.437500	-52.500000	-51.562500
-50.625000	-49.687500	-48.750000	-47.812500	-46.875000
-45.937500	-45.000000	-44.062500	-43.125000	-42.187500
-41.250000	-40.312500	-39.375000	-38.437500	-37.500000
-36.562500	-35.625000	-34.687500	-33.750000	-32.812500
-31.875000	-30.937500	-30.000000	-29.062500	-28.125000
-27.187500	-26.250000	-25.312500	-24.375000	-23.437500
-22.500000	-21.562500	-20.625000	-19.687500	-18.750000
-17.812500	-16.875000	-15.937500	-15.000000	-14.062500
-13.125000	-12.187500	-11.250000	-10.312500	-9.375000
-8.437500	-7.500000	-6.562500	-5.625000	-4.687500
-3.750000	-2.812500	-1.875000	-0.937500	

Table 2: Acceptable latitude values

-89.27665	-88.33975	-87.39729	-86.45353	-85.50930
-84.56487	-83.62028	-82.67562	-81.73093	-80.78618
-79.84142	-78.89662	-77.95183	-77.00701	-76.06219
-75.11736	-74.17252	-73.22769	-72.28285	-71.33799
-70.39314	-69.44830	-68.50343	-67.55857	-66.61371
-65.66885	-64.72399	-63.77912	-62.83426	-61.88939
-60.94452	-59.99965	-59.05478	-58.10991	-57.16505
-56.22018	-55.27531	-54.33043	-53.38556	-52.44069
-51.49581	-50.55094	-49.60606	-48.66119	-47.71632
-46.77144	-45.82657	-44.88169	-43.93681	-42.99194
-42.04707	-41.10219	-40.15731	-39.21244	-38.26756
-37.32268	-36.37781	-35.43293	-34.48805	-33.54317
-32.59830	-31.65342	-30.70854	-29.76366	-28.81879
-27.87391	-26.92903	-25.98415	-25.03928	-24.09440
-23.14952	-22.20464	-21.25977	-20.31489	-19.37001
-18.42513	-17.48025	-16.53537	-15.59050	-14.64562
-13.70074	-12.75586	-11.81098	-10.86610	-9.921225
-8.976346	-8.031467	-7.086589	-6.141711	-5.196832
-4.251954	-3.307075	-2.362196	-1.417318	-0.4724393
0.4724393	1.417318	2.362196	3.307075	4.251954
5.196832	6.141711	7.086589	8.031467	8.976346
9.921225	10.86610	11.81098	12.75586	13.70074
14.64562	15.59050	16.53537	17.48025	18.42513
19.37001	20.31489	21.25977	22.20464	23.14952
24.09440	25.03928	25.98415	26.92903	27.87391
28.81879	29.76366	30.70854	31.65342	32.59830
33.54317	34.48805	35.43293	36.37781	37.32268
38.26756	39.21244	40.15731	41.10219	42.04707
42.99194	43.93681	44.88169	45.82657	46.77144
47.71632	48.66119	49.60606	50.55094	51.49581
52.44069	53.38556	54.33043	55.27531	56.22018
57.16505	58.10991	59.05478	59.99965	60.94452
61.88939	62.83426	63.77912	64.72399	65.66885
66.61371	67.55857	68.50343	69.44830	70.39314
71.33799	72.28285	73.22769	74.17252	75.11736
76.06219	77.00701	77.95183	78.89662	79.84142
80.78618	81.73093	82.67562	83.62028	84.56487
85.50930	86.45353	87.39729	88.33975	89.27665

D Lambert Conformal Domain Example

This section describes how to compute the values for the run domain and param domain sections on a Lambert conformal projection.

STUB!

DRAFT

E Mercator Domain Example

This section describes how to compute the values for the run domain and param domain sections on a Mercator projection.

STUB!

DRAFT

F UTM Domain Example

This section describes how to compute the values for the run domain and param domain sections on a UTM projection.

DRAFT

G `configure.lvt`

This is a sample `configure.lvt` file used for compiling LVT on the NCCS discover machine (Linux, 64bit with intel compiler)

```
FC           =      mpif90
FC77        =      mpif90
LD          =      mpif90
CC          =      mpicc
AR          =      ar ru
INC_NETCDF  = /home/user/lib/netcdf-3.6.3/include/
LIB_NETCDF  = /home/user/lib/netcdf-3.6.3/lib/
LIB_ESMF    = /home/user/esmf/lib/lib0/Linux.intel.64.mpi.default/
MOD_ESMF    = /home/user/mod/mod0/Linux.intel.64.mpi.default/
CFLAGS      = -c -DIFC
FFLAGS77    = -c -O0 -nomixed_str_len_arg -names lowercase \
              -convert big_endian -assume byterecl -DHIDE_SHR_MSG \
              -DNO_SHR_VMATH -DIFC -I$(MOD_ESMF) -DSPMD -DUSE_INCLUDE_MPI

FFLAGS      = -c -g -u -traceback -fpe0 -nomixed_str_len_arg \
              -names lowercase -convert big_endian -assume byterecl \
              -DHIDE_SHR_MSG -DNO_SHR_VMATH -DIFC -I$(MOD_ESMF) \
              -I$(INC_NETCDF) -DUSE_INCLUDE_MPI

LDFFLAGS    = ../lib/w3lib/libw3.a ../lib/read_grib/readgrib.a \
              ../lib/grib/griblib.a -lmpi -L$(LIB_NETCDF) -lnetcdf \
              -L$(LIB_ESMF) -lesmf \
              -lstdc++ -limf -lm -lgcc -lgcc_s -lrt
```

References

- [1] W. Sawyer and A. da Silva. Protex: A sample fortran 90 source code documentation system. Technical report, NASA GMAO, 1997. DAO Office Note 97-11.

DRAFT